Integrating Lecture Recordings with Social Networks

Patrick Fox, Johannes Emden, Nicolas Neubauer and Oliver Vornberger

Institute of Computer Science
University of Osnabrück
Germany, 49069 Osnabrück
E-Mail: \{pfox | jemden | nineubau | oliver\}@uos.de

Abstract

This paper describes an approach to integrate lecture recordings, based on the virtPresenter framework, within social networks. A lot of social networks have recently begun to provide application programming interfaces (APIs) for external applications, allowing access to users’ profile information and their friendship relations. The project “social virtPresenter” combines the resulting social graph with the existing lecture recordings. Thus, “social virtPresenter” provides a basis for a new collaborative multimedia learning process.

1. Introduction

Since the beginnings of university lectures, knowledge transfer has consisted of the classic lecture scheme. A lecturer writes content on the blackboard and the students take notes to use them for further studies at home.

Nowadays, this approach has changed a lot with the use of multiple technologies which have led to more comfort for today’s students. Firstly, presentation tools such as PowerPoint offered the lecturer the opportunity to display his contents with the help of a projector. Then came the recording of lectures, eliminating the need to take notes. Meanwhile, techniques have been developed to combine lecture recordings and presentation slides, in order to optimally prepare lecture content and make it publicly available [5].

With “social virtPresenter” we take the next step forward and link the lecture recording system to social networks. While the process of providing lecture recordings in many different ways, e.g., the integration with an LMS [4], is not new at all, our approach - the integration within a social network - focusses a lot more on the user, his experience and especially the inter user experience based on the social graph. Thus, our goal is to create a socially fully integrated learning environment.

In the following, we would like to illustrate the opportunities given by our test implementation in the social network Facebook [1], linking lecture recordings with a modern communication platform.

2. Fundamentals

Before presenting our work we will outline the basic concepts of social networks and prerequisites of lecture recordings.

2.1. Social Networks

Social networks are an essential part of the Web 2.0 movement. A social network offers users two main components. On the one hand, a user can create a profile in which he can publish private or professional details or personal interests. On the other hand, a user can maintain his personal contact list and connect with other participants to establish a social network. These connections between many users form the social graph. It provides the basis for users to communicate with each other, either through direct messages or indirectly through notifications that are sent when a user of the social network performs certain actions, such as the publishing of photos. In social networks, users build a web community which provides fast information exchange.

Recently, social networks have begun to offer an API to developers to access profile data and the social graph of users in order to create their own applications hereon.
2.2. Lecture Recordings

Now that the concepts of social networks and their possibilities for external applications have been described, we will proceed with the prerequisites for our content.

First of all the content has to be in a digital form that allows us to present it in a way compatible with web application. In our case we took advantage of the virtPresenter framework to generate the video stream and the lecturers’ presentation material, mostly PowerPoint slides, to be streamable within a Flash based player. Actually the framework automatically generates the player, too. As we intend to focus on the integration of the actual lecture recordings, there is no further discussion of the framework. Generally speaking, the process of pre-processing the already web-enabled video and its meta-data looks as illustrated in the diagram in Figure 2.

![Figure 2. Data Flow](image)

The process starts with indexing or copying the actual video data while the required meta-data has to be gathered. These should contain at least a valid title, a course the lecture belongs to and the runtime. Moreover, useful information could be a slide or chapter index, searchable content of the slides full text, and thumbnails for those. All this information should be persisted in a single place, either directly attached to the video data or centrally stored, e.g., in a database. We decided to take the latter for it is easier to connect the general metadata with the social data that will be gathered while the application is being used.

3. Combining Social Networks with Lecture Recordings

Now, as we have determined the fundamentals, prerequisites and possibilities of our platform, the social network, the content and the video and meta-data of the lectures, we will present an overview of the concepts and implementation details we decided to use in our “social virtPresenter” application that combines lecture recordings and social networking on the basis of the Facebook platform.

3.1. Content presentation

When it comes to content presentation we distinguish between three levels, where social meta-information can be used to enhance our content and make content navigation social: user generation, presentation of the content catalogue and the specific view of one item or lecture in particular. As these levels are not distinct in the concepts used, they can be applied to the other levels in most cases, too.

Firstly, we will look at the generation of new users. Most of the social networks that allow third party applications provide a single space where an “install this application”-page can be stored. Moreover it is common that rudimentary profile and friendship information can be read by the application on this page. In a traditional application you could only guess what might convince the current user to install your application. In most cases you will see something like a “Our users like ...”-section where arbitrary content or, at best, content with a high ranking by other users is presented. As we are in a social context, we know which friends of the current user are already using our application. This implies a certain correlation between the user’s friends’ preferences and what the user himself likes, which enables us to provide a much better selection of content that makes our application - even in this first and earliest stage of usage - personal to the user.

While the same concept of personalisation can be applied to the second level, the content catalogue navigation and presentation, user awareness should be focused on as a concept of more relevance in this level. For further information on how social meta-data can be leveraged to socially present the content catalogue, see Section 3.5.

Also in the third level, which deals with a specific view on one chosen lecture, the social graph can be used to display interesting statistics on the usage of a user’s friends. For example, it is possible to see which of his friends have already watched the lecture he chose. He would therefore have someone to discuss the topic with and possibly question. Especially in this level it makes sense to go beyond the first level of the social graph, to see who else has watched the lecture and may be a fellow student, which a user has not yet added as a friend in the network. Beside observing the past - “Your friends watched ...” - there should also be real time information on the activities of a user’s friends available, e.g., who I could join watching a lecture (also see Section 3.5).

Naturally the usage of social meta-data should not be limited to only informing users. This can only be the basis for a social application. Therefore these concepts lead to the actual interaction of the users, which is covered in the following sections.
3.2. User Communication

As stated above, users should not only be able to see other users’ actions but also take part in those by interacting. Especially in an application with a learning-topic, sharing information is extremely important. In the area of user communication our application is separated into three layers and one meta-layer.

The most general part are the bulletin boards. These exist separately from each other while each course has one of those. Questions or comments posted here are not referring to a specific lecture or even a part of one but to the course itself. For this place, we thought of questions about quizzzes and tests students have to take or the ability to look for a learning partner (also see Section 3.4).

A little more detail is added to the layer of the comment system. In our application each lecture recording has its own section, where users can ask questions directly, referring not only to the recording at hand, but even to a specific part of the lecture. Of course comments do not need to be questions; they can be anything that might be interesting for the lecture. Thanks to the current position of the user being attached while writing a comment, the lecture becomes indexed by its viewers in a tag-like way and context specific discussions are made possible. Being updated with AJAX technology and new comments being pushed to all watching users, new contributions to the discussions can be received or written during the video without it needing to be stopped.

As the bulletin boards and commentary systems are available at all times for following viewers, the integrated chat is the only volatile communication medium in our application. In combination with the Social Scrubber, described in Section 3.5.2, short chat messages can be sent globally to all current viewers of a lecture. Below the user’s picture, representing the sender of the message, a speech bubble will be shown for all other viewers. This creates a shout or whisper mode, that provides means to general chatting as well as short questions that can be answered by the other viewers with a word or a short sentence, e.g., “Can you read what’s written in the second line on the board?” - “It says ’runtime analysis’”.

In order for new comments, contributions to the boards or the user’s activities to be noticed by other users, we provide a last automated meta-layer of indirect communication. That is the notification system. Interesting actions or activities of a user, like starting the application, watching a specific lecture or posting a new comment are announced in different ways. On Facebook, there are multiple possibilities to notify other users about their friends’ activities and the so called news stream allows us to publish a user’s actions on his own profile page, e.g., “Max is watching Computer Science 101”. By providing these three layers of communication and therewith also a knowledge exchange, we have enhanced the user’s learning activity while watching a lecture recording to make it a social and more efficient experience.

3.3. Collaboration

When thinking of knowledge exchange on the internet, Wikipedia and the concept of collaborative information sharing comes to the mind. The same concept is behind collaboration in “social virtPresenter”. Similarly to the comment system and the bulletin boards (see Section 3.2) each lecture and also each course possesses a wiki-like page which can be read but also changed and complemented by all users. For editing we abandoned a complicated markup language by design and only allow simple structural elements such as headings and enumerations. This offers users the possibility to publish their thoughts as simple and quickly as possible without thinking longer about formatting conventions than their actual contribution. Beside the information that can be added by the users, e.g., “Where does the weekly study group meet?”, the lecturer or his assistants can make contributions. Regarding this, we thought of adding literature and links to old tests or exercises.

3.4. Virtual Study Groups

In our application users have the possibility to form virtual study groups with their friends and other users of the social network. A study group is assigned to a course and can be created by any user. By defining a password and sharing it with the users he wants to join the creator grants access to his group. Members of a study group share a board for discussions, a calendar and advanced functions for notification. Users can write messages that are addressed to all members of their study group. We combined the concept of virtual study groups with different aspects of user awareness that we describe in Section 3.5.

3.5. User Awareness

An essential aspect of social applications is the possibility for the user to interact at any time with other users. This requires real time information about the status of other users: the more one knows about the current activities of another user the more specific they can interact.

3.5.1. Who’s online. The content catalogue and the list of lecture recordings include a small list of avatars for each entry, in which those users are listed who are currently watching a lecture recording (see Figure 3). These linked profile photos give the user the opportunity to directly reach the profile of another user, as well as to start the recording they are currently watching and to get in touch with them, e.g., via the chat feature, described in Section 3.2.
3.5.2. Social Scrubber. While playing a lecture recording the profile photos of users who are watching the same recording are displayed again (see Figure 4). They appear below the timeline according to their current position in the lecture recording.

Thus, everyone knows how far other users are in the recording. In combination with the chat feature, described in Section 3.2, this creates a virtual lecture hall for the user, where he can meet other users watching the same recording, even if they are at different positions within the recording.

3.5.3. Footprints. The concept of footprints known from the internet browsers suggests that already visited text links should be highlighted to enhance navigation on web sites. Footprints are useful to identify the relevant parts of lecture recordings without having watched the entire recording. Applied to the visualisation of time-based contexts, footprints are coupled to the timeline [5]. The underlying concepts of highlighting footprints of lecture recordings were developed in a joint research by the University of Osnabrück and the University of Pittsburgh [3].

The intensity of the colour highlighting a certain part is an indicator for the number of views in comparison to other parts. The most watched parts are marked with very intense colours while less intense colours mark fewer watched parts.

We integrated different options with our application giving the user the ability to decide whether to visualise his own footprints, his virtual study group’s footprints or all other footprints. Furthermore, we added the feature of a dual-view that shows the former and the latter visualisation in comparison.

3.6. Data Flow and System Architecture

Although this paper is intended to provide a conceptual overview of the different aspects we decided to implement in our application, some insight into the resulting system architecture and the general data flow should be given.
which we would consider a way of direct communication. Technically however, chat messages and notifications are sent with Facebook, acting as a gateway. An exception to this concept of relaying all requests through the Facebook servers is the streaming of the video by using the Flash based player. In this case the Facebook servers are not used to negotiate requests. Once our application has delivered the Flash player, all streamed data is directly loaded from the virtPresenter framework back-end where the actual video data is stored.

Thus, the system’s architecture is divided into the user layer with its direct communication, the Facebook layer for negotiating requests and as user meta data store, the “social virtPresenter” layer, with all current application controllers, interfaces and local user data as well as enriched video meta data, and lastly the virtPresenter framework back-end as initial video meta data source and video data store.

4. Conclusion and Future Work

From a user’s perspective the virtPresenter lecture recording framework initially consisted of a stand-alone player. This paper presents a solution to the integration of a multimedia player within a social network, such as the Facebook platform. Thus, the properties of a social network are coupled with lecture recordings. It is shown how the existing repository of lecture recordings can be presented to a user of a social network and what options for interaction, based on the social graph, can be made available. An essential part are concepts of user awareness such as the Social Scrubber and the footprints as well as the discussion boards in the context of lecture recordings, study groups and entire courses. In summary, we demonstrated an approach to extend lecture recordings with social features and user-generated content. The developed application currently exists in an early publicly available stage. It is currently being tested by students of the University of Osnabrück.

In further research the user data currently being generated should be analysed to prove the acceptance and benefits of the concepts demonstrated. Moreover, the virtPresenter project has recently joined the OpenCast Initiative [6] proposed by the University of Berkeley and an integration of “social virtPresenter” with Google’s OpenSocial [2] is planned.

Acknowledgment

The authors want to recognize the work of Roland Tapken, Andreas Rickling, Marcel Trame and Christian Hüser who participated in the project group “social network applications” at the University of Osnabrück.

References